

Applicant Initiated Interview Request Form

SYM 0002

Application No.: 10/756,894

Conf. No. 3627

First Named Applicant: West

Examiner: Thuy Chan Dao

Art Unit: 2192

Status of Application: Pending

Tentative Participants:

(1) Thuy Chan Dao

(2) Paul Davis

Proposed Date of Interview: Tues., Dec 16, 2008

Proposed Time: 2:00 (AM/PM) (**Eastern Time**)

Type of Interview Requested:

(1) ☒ Telephonic (2) ☐ Personal (3) ☐ Video Conference

Exhibit to be Shown or Demonstrated: ☐ Yes ☒ No

If yes, provide brief description: _____

Issues to be Discussed

Issues (Rej., Obj., etc.)	Claims/ Fig. #s	Prior Art	Discussed	Agreed	Not Agreed
(1) <u>Rejections</u>	<u>50-52</u>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Brief Description of Argument to be Presented:

How the amended claims overcome the prior art. See attached claims 50-52 for discussion.

An interview was conducted on the above-identified application on _____.

NOTE: This form should be completed by applicant and submitted to the examiner in advance of the interview (See MPEP § 713.01)

This application will not be delayed from issue because of applicant's failure to submit a written record of this interview. Therefore, applicant is advised to file a statement of the substance of this interview (37 CFR 1.133(b)) as soon as possible.

Paul Davis
Applicant/Applicant's Representative Signature

Examiner/SPE Signature

PAUL DAVIS
Typed/Printed Name of Applicant or Representative

29294
Registration Number, if applicable

10/756,894 CLAIMS FOR DISCUSSION:

50. (currently amended) A method for transforming test cases that are converted from a source test script to an abstract representation and storing abstract representation of test cases into a database system data store, comprising:

storing test cases in abstract representations to generate test cases in any target environment script format to provide interoperability between automation tools and cross environment portability of test cases;

importing at least one test case[[s]] written in one or more scripting language[[s]];

~~using semantic analysis to convert test cases to an abstract representation that includes application state, external interaction sequences and input data without changing or deleting an original test case based on an application object model~~
converting the at least one test case to an abstract representation using semantic analysis without changing or deleting an original test case based on an application object model, the abstract representation having at least three separate components including an application state, external interaction sequences and input data, where the application object mode is a metadata representation for a modeling application under test and includes components selected from application object type definitions for application objects, attribute definitions for each application object type, definitions of methods and events that are supported by each application object type and definitions of effects of events on an application state[[.]] and

storing the abstract representation into a database; and

generating at least one platform-specific test script based on the abstract representation stored in the database using a platform-specific mapping, the platform-specific mapping including a language mapping and an environmental mapping.

51. (currently amended) A system for transforming test cases that are converted from a source test script to an abstract representation and storing abstract representation of test cases into a data store, comprising:

a processor for importing test cases written in one or more scripting languages;

a database for storing abstract representations;

~~logic that uses semantic analysis to convert test cases to an abstract representation that includes application state, external interaction sequences and input data without changing or deleting an original test case based on an application object model~~ converts the at least one test case to an abstract representation using semantic analysis without changing or deleting an original test case based on an application object model, the abstract representation including an application state, external interaction sequences and input data, where the application object model is a metadata representation for modeling application under test and includes components selected from application object type definitions for application objects, attribute definitions for each application object type, definitions of methods and events that are supported by each application object type and definitions of effects of events on an application state, ~~the logic using environment mappings providing platform independence of test cases and test scripts are generated for multiple test execution environments without changing or deleting an original test case, the test cases being recombined and modified using external rules to combine and modify components of the abstract representation of test cases into new scripts~~ the logic generating at least one platform-specific test script based on the abstract representation stored in the database using a platform-specific mapping, the platform-specific mapping including a language mapping and an environmental mapping.

52. (currently amended) A computer system for transforming test cases that are converted from a source test script to an abstract representation and storing abstract representation of test cases into a data store, comprising:

a processor; and

a database for storing abstract representations;

a memory coupled to the processor, ~~the memory storing test cases in abstract representations to generate test cases in any target environment script format to provide interoperability between automation tools and cross-environment portability of test cases, importing test cases written in one or more scripting languages and uses semantic analysis to convert test cases to an abstract representation that includes application state, external interaction sequences and input data without changing or~~

~~deleting an original test case based on an application object model, where the application object model is a metadata representation for modeling application under test and includes components selected from application object type definitions for application objects, attribute definitions for each application object type, definitions of methods and events that are supported by each application object type and definitions of effects of events on an application state .~~ the at least one test case that is converted to an abstract representation through semantic analysis without changing or deleting an original test case based on an application object model, the abstract representation including an application state, external interaction sequences and input data, where the application object model is a metadata representation for modeling application under test and includes components selected from application object type definitions for application objects, attribute definitions for each application object type, definitions of methods and events that are supported by each application object type and definitions of effects of events on an application state, the memory storing at least one platform-specific test script generated based on the abstract representation stored in the database by use of a platform-specific mapping, the platform-specific mapping including a language mapping and an environmental mapping.